PATENT

MS302986.01/MSFTP494US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re patent application of:			
Applicant(s):	Lingan Satkunanathan, *et al*	Examiner:	Evens J. Augustin
Serial No:	10/723,324	Art Unit:	3621
Filing Date:	November 26, 2003		

Title: REAL-TIME LICENSE ENFORCEMENT SYSTEM AND METHOD

Mail Stop Appeal Brief – Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Dear Sir:

Appellants' representative submits this brief in connection with an appeal of the above-identified patent application. A credit card payment form is filed concurrently herewith in connection with all fees due regarding this appeal brief. In the event any additional fees may be due and/or are not covered by the credit card, the Commissioner is authorized to charge such fees to Deposit Account No. 50-1063 [MSFTP494US].

01/04/2006 MBINAS 00000017 10723324
01 FC:1402 500.00 OP

I. Real Party in Interest (37 C.F.R. §41.37(c)(1)(i))

The real party in interest in the present appeal is Microsoft Corp., the assignee of the present application.

II. Related Appeals and Interferences (37 C.F.R. §41.37(c)(1)(ii))

Appellant, appellants' legal representative, and/or the assignee of the present application are not aware of any appeals or interferences which may be related to, will directly affect, or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. Status of Claims (37 C.F.R. §41.37(c)(1)(iii))

Claims 1-4, 6-18, 20-33, and 35-43 are currently pending in the subject application and are presently under consideration. Claims 1-4, 6-18, 20-33, and 35-43 stand rejected by the Examiner. Claims 5, 19 and34 have been cancelled. The rejection of claims 1-4, 6-18, 20-33, and 35-43 is being appealed.

IV. Status of Amendments (37 C.F.R. §41.37(c)(1)(iv))

No amendments have been entered subsequent the Final Office Action dated August 3, 2005.

V. Summary of Claimed Subject Matter (37 C.F.R. §41.37(c)(1)(v))

A. Independent Claim 1

Independent claim 1 recites a license enforcement system that comprises a monitoring component that determines a current number (M) of users logged on to an application under a license, an enforcement component that dynamically takes corrective action if M exceeds an authorized number (N) of users permitted under the license, and a validation component that periodically checks stored license data to ensure it has not been corrupted. (*See e.g.*, page 7, line 21 – page 8, line 6).

B. Independent Claim 11

Independent claim 11 recites a license enforcement system that comprises a monitoring component that monitors use of a licensed application by one or more users in accordance with a license agreement, an enforcement component that takes corrective action upon notification by the monitoring component that there has been a violation of the license agreement, and a validation component that periodically checks stored license data to ensure it has not been corrupted. (*See, e.g.*, page 7, line 21 – page 8, line 6).

C. Independent Claim 20

Independent claim 20 recites a licensing system that comprises a license store that receives license data from a license component associated with a licensed application, a monitoring component that monitors license data and utilization of the application to determine license violations, an enforcement component that takes corrective action in response to notification of a license violation by the monitoring component, and a validation component that periodically checks stored license data to ensure it has not been corrupted. (*See e.g.*, page 7, line 21 – page 8, line 6).

D. Independent Claim 29

Independent claim 29 recites a method for enforcing licensing agreements that comprises monitoring license data in a data store and use of a licensed application, determining license violations, initiating corrective action in response to detection of a license violation, and checking stored license data on a periodic basis to ensure that the data has not been corrupted. (*See e.g.*, page 8, lines 16-30).

E. Independent Claim 40

Independent claim 40 recites a method of adding licenses to a computer system that comprises activating a license component, installing the license component on a computer, wherein installing the license component includes storing license data in a license store, and checking stored license data on a periodic basis to ensure that the data has not been corrupted. (*See e.g.*, page 17, line 25 – page 18, line 2).

VI. Grounds of Rejection to be Reviewed (37 C.F.R. §41.37(c)(1)(vi))

A. Claims 1-4, 6-18, 20-33, and 35-43 stand rejected under 35 U.S.C. §102(e) as being unpatentable over Christiano (5,671,412).

VII. Argument (37 C.F.R. §41.37(c)(1)(vii))

A. Rejection of Claims 1-4, 6-18, 20-33, and 35-43 Under 35 U.S.C. §102(e)

Claims 1-4, 6-18, 20-33, and 35-43 stand rejected under 35 U.S.C. §102(e) as being anticipated by Christiano (5,671,412). Reversal of this rejection is respectfully requested for at least the following reasons. Christiano fails to teach or suggest all limitations recited in the subject claims.

> For a prior art reference to anticipate, 35 U.S.C. §102 requires that "***each and every element*** as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *In re Robertson*, 169 F.3d 743, 745, 49 USPQ2d 1949, 1950 (Fed. Cir. 1999) (*quoting Verdegaal Bros., Inc. v. Union Oil Co.*, 814 F.2d 628, 631, 2USPQ2d 1051, 1053 (Fed. Cir. 1987)). (emphasis added)

As described in previous correspondence, the subject invention generally relates to a license management and enforcement system that regularly checks license data contained in the system. This novel feature is neither taught nor suggested in Christiano. In particular, Christiano does not teach or suggest ***a validation component that periodically checks stored license data to ensure that the data has not been corrupted*** as recited in independent claim 1 (and similarly in independent claims 11, 20, 29, and 40).

Christiano discloses a license management system for software applications that stores and manages license data using a client-server model. A license server stores software application license data, which is used to grant or deny license requests for various applications from client systems. The Examiner contended that "defined

intervals" is not defined in the specification (*See* Advisory Action dated October 17, 2005), but Appellants' representative respectfully disagrees. The Examiner further contended that checking "from time to time" equates to ***periodically*** checking license data, as in the claimed invention. (*See* Office Action dated August 3, 2005). The Examiner previously cited portions of the reference that address checking the status of the network between a client and server (*See* col. 3, lines 18-20) and determining the availability of licenses (*See* col. 17 lines 55-64), but these sections of the reference do not disclose checking ***stored license data*** at all, much less ***periodically*** checking.

The license server disclosed in Christiano performs diagnostic checking of the license store data upon receiving a license request from a user (col. 10, lines 53-55 and col. 19, lines 17-21), upon initialization of the license store (col. 11, lines 12-24), or in response to a request from a client system to perform diagnostic checks (col. 21, lines 19-28 and col. 22, lines 29-33). Because the license diagnostics are only performed in response to system events that do not occur with any specified frequency, the verification clearly cannot be construed to be ***periodic***, as recited specifically in claims 1, 11, 20, and 40. One of ordinary skill in the art on reading the word "periodic" would perceive it to mean: occurring or recurring at regular intervals, which is synonymous with ***defined intervals*** as recited in claim 29. Although the reference discloses that the server has the ability to check license data in some manner, it does not disclose that these verifications occur with any temporal regularity. Accordingly, it would be more appropriate to classify such actions as *aperiodic* because they lack regularity and defined frequency. In contrast, the instant specification teaches checking of license data with a fixed frequency (*e.g.* daily), which is a periodic event. (*See* page 9, lines 1-3). Consequently, Christiano is silent with regard to a ***validation component that periodically checks stored license data to ensure that the data has not been corrupted***, as recited in the subject claims.

Even if it were incorrectly assumed that Christiano does disclose that actions are performed periodically, the reference is silent with regard to verifying that ***the data has not been corrupted***. The reference discloses checking stored information as a method for deciding whether a license should be provided to a requesting client, and the verification performed by the license server consists of examining the "status" of the license to determine whether it is "granted" or "denied." (*See* col. 25, lines 51-54). Additionally,

the failure-checking in the system is limited to verifying that the license management system, meaning the components of the system and not the license data, itself is functioning properly. (*See* col. 18, lines 30-33). The cited portion of the reference (*see* Office Action dated April 26, 2005) fails to actually verify the license data and only checks the *key* associated with the license to determine if there have been any changes in the stored license. (*See* col. 19, lines 18-21). The reference also discloses checking the *header* information *associated* with the license, but again does not specifically verify that the ***license data*** itself ***has not been corrupted.*** Although the system disclosed in the reference does have limited verification capabilities, none deals specifically with the idea of checking for data corruption. Data corruption occurs when at least one bit of data is altered inadvertently and renders the data unusable. The validation component of the claimed invention, however, validates the license data against a backup license store database ***to ensure that the data has not been corrupted.*** As shown in the instant specification, checking for data corruption involves the comparison of the licenses in the backup database against those stored in the license store. (*See* page 9, lines 1-5). By comparing the current and backup license data, the claimed system can identify inconsistencies that require further actions. Given the absence of such disclosure, it is further evident that the reference does not teach or suggest ***a validation component that periodically checks stored license data to ensure that the data has not been corrupted.***

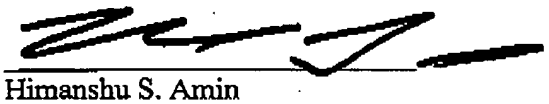
Accordingly, it is readily apparent that the reference does not anticipate the subject invention; therefore, it is respectfully requested that the rejection of claims 1, 11, 20, 29, and 40 (and 2-4, 6-10, 12-18, 21-28, 30-39, and 41-43 which depend there from, respectively) be reversed.

B. Conclusion

For at least the above reasons, the claims currently under consideration are believed to be patentable over the cited references. Accordingly, it is respectfully requested that the rejections of claims 1-4, 6-18, 20-33, and 35-43 be reversed.

If any additional fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063 [MSFTP494US].

Respectfully submitted,
AMIN & TUROCY, LLP



Himanshu S. Amin
Reg. No. 40,894

AMIN & TUROCY, LLP
24th Floor, National City Center
1900 East 9th Street
Telephone: (216) 696-8730
Facsimile: (216) 696-8731

VIII. Claims Appendix (37 C.F.R. §41.37(c)(1)(viii))

1. A license enforcement system, comprising:

a monitoring component that determines a current number (M) of users logged on to an application under a license;

an enforcement component that dynamically takes corrective action if M exceeds an authorized number (N) of users permitted under the license; and

a validation component that periodically checks stored license data to ensure that the data has not been corrupted.

2. The system of claim 1, wherein corrective action includes providing a warning to one or more users.

3. The system of claim 1, wherein the corrective action comprises denying one or more users access to a license target, the license target including one of the application, a system and a device.

4. The system of claim 1, further comprising a license store for securely storing license data.

5. (Cancelled)

6. The system of claim 1, further comprising a backup store that contains copies of stored license data.

7. The system of claim 6, wherein the validation component restores corrupted stored license data by copying data in the backup store to the license store.

8. The system of claim 1, further comprising an oversight component that observes the monitoring component to ensure that it is operating.

9. The system of claim 8, wherein the oversight component restarts the monitoring component if is stops operating.

10. The system of claim 9, wherein the oversight component shuts down the application if monitoring component cannot be restarted.

11. A license enforcement system comprising:

a monitoring component that monitors use of a licensed application by one or more users in accordance with a license agreement;

an enforcement component that takes corrective action upon notification by the monitoring component that there has been a violation of the license agreement; and

a validation component that periodically checks stored license data to ensure that the data has not been corrupted.

12. The system of claim 11, wherein the monitoring component monitors user logins.

13. The system of claim 11, wherein the monitoring component monitors hardware associated with the execution of an application.

14. The system of claim 11, wherein the monitoring component monitors use of other applications being executed concurrently with the licensed application.

15. The system of claim 11, wherein the corrective action corresponds warning one or more users of the licensing agreement violation.

16. The system of claim 11, wherein the corrective action corresponds to shutting down the licensed application.

17. The system of claim 11, further comprising an oversight component to observe the operation of the monitoring component and restart the component if it fails to operate properly.

18. The system of claim 11, further comprising a license store for storing licenses and data associated therewith.

19. (Cancelled)

20. A licensing system comprising:

a license store that receives license data from a license component associated with a licensed application;

a validation component that periodically checks stored license data to ensure that the data has not been corrupted;

a monitoring component that monitors license data and utilization of the application to determine license violations; and

an enforcement component that takes corrective action in response to notification of a license violation by the monitoring component;

21. The system of claim 20, wherein the license component comprises:

an installation component that saves license data to the license store; and

an activation component that provides a mechanism for activating the license component, wherein license data is saved to the license store by the installation component after the license is activated.

22. The system of claim 21, wherein license data includes a value indicative of the number of licenses provided by the license component.

23. The system of claim 21, wherein the license data includes license issue date and expiration date.

24. The system of claim 21, wherein the license data includes a hardware ID indicative of the hardware components of a computer system utilized to execute the licensed application and a license ID that uniquely identifies the license component.

25. The system of claim 20, further comprising a bypass component that disables the monitoring component and/or the enforcement component.

26. The system of claim 25, wherein the bypass component is a license component with a license code within a bypass range.

27. The system of claim 20, further comprising an interface component to facilitate information retrieval by a user.

28. The system of claim 27, wherein the interface is utilized by the enforcement component to provide notification to users in violation of a licensing agreement.

29. A method for enforcing licensing agreements comprising:
monitoring license data in a data store and use of a licensed application;
determining license violations;
initiating corrective action in response to detection of a license violation; and
checking stored license data at defined intervals to ensure that the data has not been corrupted.

30. The method of claim 29, wherein determining license violations comprises comparing the number of users licensed to utilize the application against the number of users actually using the application.

31. The method of claim 30, wherein the number of users actually using an application corresponds to the number of users logged onto a system.

32. The method of claim 29, wherein determining license violations comprises comparing the number of licenses to the number of devices, a violation occurring where the number of devices is greater than the number of licenses.

33. The method of claim 29, wherein determining license violations comprises monitoring computer hardware configuration and comparing it with an allowable configuration specified by the licensing agreement.

34. (Cancelled)

35. The method of claim 29, further comprising retrieving license data from a backup store and saving it to the license store if license store data has been tampered with or corrupted.

36. The method of claim 29, wherein the license data is encrypted.

37. The method of claim 29, wherein the corrective action includes warning a user of the violation.

38. The method of claim 29, wherein the corrective action includes denying one or more users access to the application.

39. A computer readable medium having stored thereon computer executable instructions for carrying out the method of claim 29.

40. A method of adding licenses to a computer system comprising:

activating a license component;

installing the license component on a computer, wherein installing the license component includes storing license data in a license store; and

checking stored license data on a periodic basis to ensure that the data has not been corrupted.

41. The method of claim 40, wherein activating a license comprises:
generating a hardware ID;
locating the license code;
transmitting the hardware ID and a license code to an activation database;
receiving an activation code from the database; and
providing the activation code to the license component.

42. The method of claim 40, wherein license data includes a number of licenses.

43. A computer readable medium having stored thereon computer executable instructions for carry out the method of claim 40.

IX. Evidence Appendix (37 C.F.R. §41.37(c)(1)(ix))

None.

X. Related Proceedings Appendix (37 C.F.R. §41.37(c)(1)(x))

None.